

MISSION OPERATIONS AND DATA SYSTEMS DIRECTORATE

Landsat 7 Mission Operations Center (MOC) to Flight Software Maintenance Element (FSME) Interface Control Document

**Revision 1
April 1997**



National Aeronautics and
Space Administration

Goddard Space Flight Center
Greenbelt, Maryland

Landsat 7 Mission Operations Center (MOC) to Flight Software Maintenance Element (FSME) Interface Control Document

Revision 1

April 1997

Prepared Under Contract NAS5-31000/HQ001057

Approved by:

W. J. Potter Landsat 7 MOC Systems Manager Code 511	Date
---	------

R. Menrad Ground System Manager Code 501	Date
--	------

E. Shell Head, Flight Software Systems Branch Code 512	Date
--	------

Goddard Space Flight Center
Greenbelt, Maryland

Landsat 7 Mission Operations Center (MOC) to Flight Software Maintenance Element (FSME) Interface Control Document

Revision 1

April 1997

Prepared Under Contract NAS5-31000/HQ001057
By Computer Sciences Corporation
CSC 10032063

Prepared by:

J. Emmerling Date
CSC

Approved by:

J. Lauderdale Date
Manager, Landsat Software Development,
CSC

Reviewed by:

C. Bauer Date
Task Lead, Landsat MOC, CSC

M. Samii Date
Technical Area Manager, Landsat 7, CSC

Quality Assured by:

S. Whisonant Date
Product Assurance Officer, CSC

Goddard Space Flight Center
Greenbelt, Maryland

Preface

This document has been baselined by the Mission Operations and Systems Development Division (MOSDD) Configuration Control Board (CCB). Proposed changes to this document shall be submitted, along with supportive material justifying the change, to the Landsat 7 Mission Operations Center (MOC) Systems Manager. Changes to this document shall be made by document change notice (DCN) or by complete revision.

Questions or comments concerning this document shall be addressed to

Landsat 7 MOC Systems Manager
Code 511
Goddard Space Flight Center
Greenbelt, Maryland 20771

Abstract

This interface control document (ICD) between the Landsat 7 Mission Operations Center (MOC) and the Flight Software Maintenance Element (FSME) defines the interface according to the Open Systems Interconnection (OSI) communications reference model. Section 1 provides an overview and background of the interface; Section 2 describes the system components of the interface and summarizes the data flow between the two. Sections 3-9 describe, respectively, the application, presentation, session, transport, network, data link, and physical layers that compose the interface.

Keywords: *Flight Software Maintenance Element (FSME), interface control document (ICD), Landsat 7, Mission Operations Center (MOC)*

Change Information Page

List of Effective Pages			
Page Number		Issue	
Title pages		Revision 1	
v through xiv		Revision 1	
1-1 through 1-3		Revision 1	
2-1 through 2-3		Revision 1	
3-1		Revision 1	
4-1 through 4-7		Revision 1	
5-1		Revision 1	
6-1		Revision 1	
7-1		Revision 1	
8-1		Revision 1	
9-1		Revision 1	
A-1 through A-2		Revision 1	
AB-1 through AB-2		Revision 1	
Document History			
Document Number	Status/Issue	Publication Date	CCR Number
511-4ICD/0196 CSC 10032063	Original	December 1996	LS7MOC-004
	Revision 1	April 1997	NA

Contents

Preface

Abstract

Section 1. Introduction

1.1	Purpose	1-1
1.2	Scope	1-1
1.3	Document Control	1-1
1.4	Document Organization	1-1
1.5	Applicable Documents	1-2

Section 2. Interface Description

2.1	General.....	2-1
2.2	Interface Description Overview.....	2-1
2.2.1	Purpose of the Interface.....	2-1
2.2.2	MOC Description.....	2-2
2.2.3	FSME Description	2-2
2.3	Data Flow Summary	2-2
2.4	Interface Security.....	2-2

Section 3. Application Layer

3.1	General.....	3-1
3.2	MOC to FSME.....	3-1

3.2.1	Dump Image Files.....	3-1
3.2.2	Star Catalog Files	3-1
3.3	FSME to MOC.....	3-1
3.3.1	Image ID Definition Files	3-1
3.3.2	Load Image Files	3-1

Section 4. Presentation Layer

4.1	General.....	4-1
4.2	MOC to FSME.....	4-1
4.2.1	Dump Image Files.....	4-1
4.2.2	Star Catalog Files	4-3
4.2.3	Load Image File Checksum Error Notification	4-3
4.3	FSME to MOC.....	4-3
4.3.1	Image ID Definition Files	4-3
4.3.2	Load Image Files	4-5

Section 5. Session Layer

5.1	General.....	5-1
5.2	MOC to FSME Transmissions	5-1
5.3	FSME to MOC Transmissions	5-1

Section 6. Transport Layer

Section 7. Network Layer

Section 8. Data Link Layer

Section 9. Physical Layer

Figures

2-1	ISO/OSI Interface Reference Model	2-3
4-1	Example of a Dump Image File	4-3
4-2	Example of an Image ID Definition File	4-5
4-3	Example of a Load Image File.....	4-7

Tables

4-1	Dump Image File Format	4-1
4-2	Dump Abstract Record Format	4-1
4-3	Dump Identification Record Format.....	4-2
4-4	Dump Data Descriptor Record Format	4-2
4-5	Dump Data Record Format.....	4-2
4-6	Image ID Definition File Format	4-4
4-7	Image ID Definition Header Record Format.....	4-4
4-8	Image ID Definition Data Record Format	4-4
4-9	Load Image File Format.....	4-5
4-10	Abstract Record Format	4-6
4-11	Identification Record Format	4-6
4-12	Data Descriptor Record Format.....	4-7
4-13	ASCII Data Record Format	4-7

Appendix A. File-Naming Conventions

Abbreviations and Acronyms

Section 1. Introduction

1.1 Purpose

This interface control document (ICD) defines the interface between the Landsat 7 Mission Operations Center (MOC) and the Landsat 7 Flight Software Maintenance Element (FSME).

1.2 Scope

In this document, networking activities are divided into two groups: user-oriented or application services and communications-oriented or transport services. User services are concerned with the formatting and interpretation of data; transport services deal with the actual transmission of the information from one system to another.

There are seven layers within this architecture structure. The upper three layers (application, presentation, and session) are associated with user services and encompass the protocols necessary to allow two dissimilar applications or operating systems to understand each other and communicate. The fourth layer (transport) isolates the upper layers from the detailed workings of the lower, network-dependent layers. It provides for reliable data transmission regardless of the nature or reliability of lower layers. The lower three layers (network, data link, and physical) are hardware specific and encompass the protocols used to interface the data communications network with the two processors exchanging information.

To the extent possible, this ICD follows the guidelines of National Aeronautics and Space Administration (NASA) provided in a handbook for preparing ICDs for nonproject-related ground facilities (Reference 1 in Section 1.5). The guidelines have been adapted to conform to the International Standards Organization (ISO)/Open Systems Interconnection (OSI) network communications reference model.

1.3 Document Control

This document has been baselined by the Mission Operations and Systems Development Division (MOSDD) Configuration Control Board (CCB). Suggested or recommended changes to this ICD should be submitted to the Landsat 7 MOC Systems Manager, who will oversee their review by the elements affected by the changes. Changes agreed to by these elements shall be distributed under the direction of the Systems Manager.

1.4 Document Organization

Section 1 contains a statement of purpose and definition of the scope of this ICD. Information is provided regarding document maintenance, organization, and supporting documents used to develop and maintain this document.

Section 2 describes the systems and the role of the interface to which this ICD applies.

Section 3 describes the application layer, which contains most user-supplied functions, the network control program, and a network management module for interactive access to the lower layers.

Section 4 describes the presentation layer, which converts data to a common format to facilitate communication between varying systems.

Section 5 discusses the session layer, which provides services such as synchronization checkpointing and error recovery to aid the orderly flow of data.

Section 6 discusses the transport layer, which provides end-to-end data integrity and quality of service functions, and assembles and disassembles packets for the network layer.

Section 7 describes the network layer, which switches and routes data transparently between computers.

Section 8 addresses the data link layer, which transfers data units to the other end of a physical link and maintains data integrity between network nodes.

Section 9 describes the physical layer, which provides bit-stream transmission over a physical medium.

Appendix A provides the file-naming conventions. This appendix is followed by a list of abbreviations and acronyms used in this document.

1.5 Applicable Documents

1. National Aeronautics and Space Administration (NASA), Goddard Space Flight Center (GSFC), STDN No. 102.8, *Handbook for Preparing Interface Control Documents for Non-Project Related Ground Facilities*, 1981
2. --, 553-FDD-95/003R0UD0, *Landsat 7/Flight Dynamics Facility Interface Control Document*, Revision Number 0, December 1995
3. Federal Aviation Administration (FAA), *NAS Open Systems Interconnection Implementation Requirements* (Draft), 1989
4. U.S. Department of Commerce, National Institute of Standards and Technology (NIST), FIPS Pub 146/XAB, *Government Open Systems Interconnection Profile (GOSIP)*, August 1988
5. --, FIPS PB90-111212/XAB, *GOSIP User's Guide*, 1989
6. --, Special Publication 500-162, FIPS PB90-212192/XAB, *Stable Implementation Agreements for OSI Protocols*, March 1990
7. NASA, GSFC, 510-1MGD/0291, *Mission Operations Division (MOD) Interface Control Document (ICD) Guidelines*, November 1991
8. --, 511-4SRD/0395, (CSC/SD-95/6043) *Landsat 7 Mission Operations Center (MOC) System Requirements Specification*, April 1996
9. --, *Landsat 7 Detailed Mission Requirements*, Baseline, July 1995
10. --, 541-185, *Nascom Operational Local Area Network (NOLAN) and MODNET II Interface Control Document*, June 1993
11. Comer, Douglas E., *Internetworking With TCP/IP Principles, Protocols, and Architecture*. Englewood Cliffs, N.J.: Prentice Hall, 1988

Section 2. Interface Description

2.1 General

This section provides functional descriptions of both interfacing systems, identifies the types of data exchanged across the interface, and discusses interface security.

2.2 Interface Description Overview

2.2.1 Purpose of the Interface

The purpose of the interface is to provide a mechanism to update and analyze the Landsat 7 spacecraft flight software. The interface is a two-way electronic interface. The MOC sends the FSME memory dump images and complete star catalog updates. The FSME sends flight software load images and memory partition files to the MOC. The FSME is responsible for providing the MOC with updates for the generation of all general memory loads.

The FSME partitions the areas of spacecraft memory into more manageable areas for use in analyzing spacecraft memory dumps and in generating load image files. Before launch, the FSME sends the MOC a file that identifies the FSME memory partitioning. Post launch, updates to the set of FSME memory partitions may be made by sending the MOC another complete set of memory partitions. The MOC uses the memory partitions provided by the FSME to aid in storing FSME load image files and spacecraft dump files.

Before launch, the FSME sends the MOC a complete set of flight software load files that comprise the entire ground reference image that can be updated through the FSME. The FSME sends the MOC one load image file for each FSME partitioned area of spacecraft memory. After launch, and whenever a change to any area of the flight software is needed, the FSME generates the necessary changes. The changes to the spacecraft flight software arrive at the MOC as load image files. Each load image file corresponds to one of the FSME partitioned areas of spacecraft memory. The MOC packages the load image files into a format understandable by the spacecraft and sends them to a Landsat 7 ground station for uplink to the spacecraft.

The MOC receives dump images of spacecraft memory through narrowband telemetry. These dump images are packaged by the MOC into an acceptable format for use by the FSME. The MOC sends the dump images to the FSME for spacecraft memory analysis.

The Flight Dynamics Facility (FDF) is responsible for generating star catalog updates. Updates to the star catalog are provided to the spacecraft by the general memory load. Because the FSME is responsible for providing the MOC with all updates for general memory loads, these changes need to originate from the FSME. The FSME and the FDF have no direct interface; therefore, the FDF must transfer the star catalog updates to the MOC. The MOC then transfers the FDF-generated file to the FSME. The FSME will convert the FDF product to suitable star catalog update in FSME/MOC-acceptable memory update format. The FDF always supplies the MOC (and therefore the FSME) with a complete star catalog update file.

2.2.2 MOC Description

The MOC, located at the Goddard Space Flight Center (GSFC) in Greenbelt, Maryland, is the focal point for all Landsat 7 satellite operations. The MOC plans and schedules the operation of the spacecraft and its science

payload, the enhanced thematic mapper plus (ETM+). To schedule contacts with the satellite, the MOC works with multiple, existing operational resources, including NASA institutional facilities and the National Oceanic and Atmospheric Administration (NOAA). The MOC generates and validates real-time commands and stored command loads based on the conflict-free schedules. During a contact, the MOC sends the commands and loads to a ground station to be uplinked to the spacecraft. The MOC monitors the health and the status of the satellite using downlinked narrowband telemetry and analyzes the long-term performance of spacecraft subsystems.

2.2.3 FSME Description

The FSME is responsible for maintaining the flight software following orbital verification and the software development and validation facility (SDVF). Troubleshooting activities of all changes to the flight software and related tables are performed on the SDVF. The SDVF consists of the special test equipment (STE) and the Landsat 7 simulator (LSIM). The STE includes hardware models of the spacecraft processor and controls interface unit and software models of the spacecraft sensors and effectors. The LSIM is an all-software simulation of spacecraft subsystem and attitude control hardware.

2.3 Data Flow Summary

Figure 2-1 represents the data flow across the interface between the MOC and the FSME mapped into the ISO/OSI reference model.

2.4 Interface Security

The interface is implemented with commercial off-the-shelf (COTS) software that supports the File Transfer Protocol (FTP) over the Transmission Control Protocol/Internet Protocol (TCP/IP). The MOC is directly connected to the closed side of the Mission Operations and Data Systems Directorate (MO&DSD) Operational/Development Network (MODNET)/Nascom Operational Local Area Network (NOLAN). Nascom will provide connections between the MODNET/NOLAN and the FSME network for MOC and FSME electronic communications.

The Landsat 7 MOC uses a primary local area network (LAN) segment for nominal operations and a backup LAN segment for contingency purposes (e.g., LAN failure or preventive maintenance). Because of the unique subnet addresses for each LAN segment, two host names are provided, each requiring a unique IP address. Therefore, in contingency mode, the backup Landsat 7 MOC IP address is required for end-to-end communications.

The IP domain, IP addresses, and host names are provided by the Landsat 7 MOC system administrator. The account names and passwords for receiving products from the FSME are provided at the time of operations. TCP/IP file transfer sessions are initiated by the FSME for

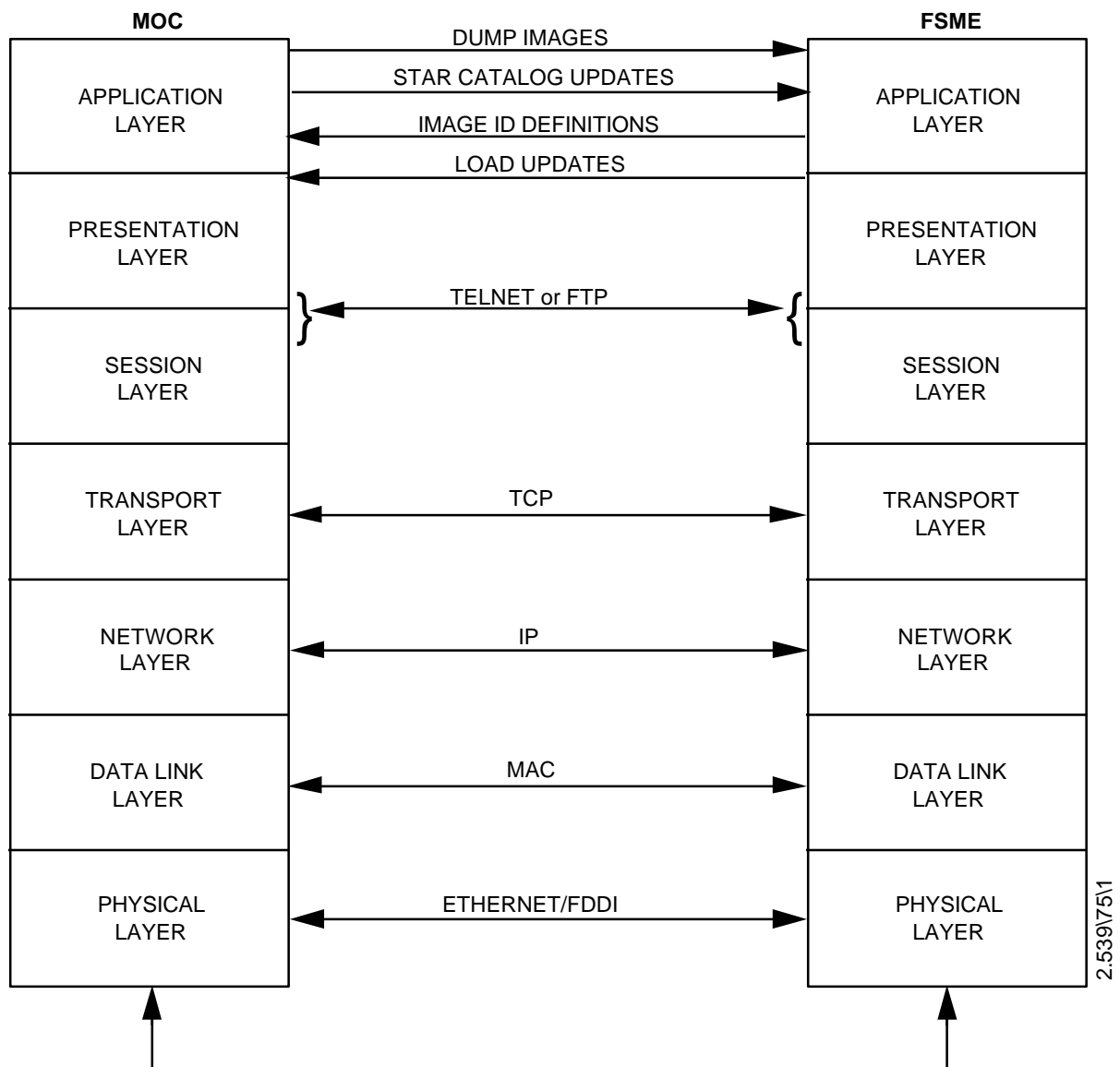


Figure 2-1. ISO/OSI Interface Reference Model

products sent to the MOC. The account name and password for transmitting products to the FSME are provided at the time of operations. TCP/IP file transfer sessions are initiated by the MOC for products transmitted to the FSME.

In case of network communication failure, the interface mechanism between the MOC and the FSME will be 3.5-inch, double-sided, high-density disks.

Section 3. Application Layer

3.1 General

The application layer defines the user data transferred across the interface to all user interface applications. This section describes the data and performance requirements levied on the exchange of information between the MOC and the FSME.

3.2 MOC to FSME

3.2.1 Dump Image Files

The Landsat 7 spacecraft transmits binary telemetry dump images to the MOC through narrowband telemetry. The MOC stores the spacecraft dump telemetry in dump image files and transfers them to the FSME. Dump image files are transmitted from the MOC to the FSME as needed.

3.2.2 Star Catalog Files

The FDF transfers updates to the star catalog as needed. The MOC acts as a pass-through node for the star catalog file. When received from the FDF, the MOC transfers the star catalog file to the FSME. All updates to the star catalog always contain the complete star catalog.

3.3 FSME to MOC

3.3.1 Image ID Definition Files

The MOC receives an initial image identifier (ID) definition file from the FSME before the Landsat 7 launch. The image ID definition file defines the FSME set of memory area partitions. The file's purpose is to define the allowable IDs for memory images that the MOC recognizes in dealing with the FSME.

3.3.2 Load Image Files

Load image files are sent by the FSME to the MOC as needed. These images are spacecraft memory data in American Standard Code for Information Interchange (ASCII) format. The FSME supplies the MOC with all the general memory loads for the Landsat 7 mission. The load image file may be a full or partial load of an area of memory specified in the image ID definition file.

Section 4. Presentation Layer

4.1 General

The following sections describe the format of the data to be transferred between the MOC and the FSME.

4.2 MOC to FSME

4.2.1 Dump Image Files

Dump images are gathered by the MOC through narrowband telemetry dumps. The dump images are converted to ASCII and stored for transfer to the FSME as needed.

The file format for the dump image file is shown in Table 4-1. Dump image files are ASCII files with ASCII header records. There are three header records: abstract record, identification record, and data descriptor record. An ASCII hexadecimal data record follows the three header records. Each record is terminated by a line-feed character (ASCII 10). Fields within records are separated by ASCII commas. Tables 4-2 through 4-4 describe the header record formats. The field widths shown in these tables do not include the terminating character. Table 4-5 describes the data record format. Figure 4-1 shows an example of the dump image file.

The file-naming convention for the dump image file is provided in Appendix A.

Table 4-1. Dump Image File Format

Record Number	Record Type	Description
1	Abstract record	Intended to describe the functional purpose of the file and other information; in actuality, always blank.
2	Identification record	Contains basic information that uniquely identifies this file; the information includes the creation date, version number, source, and comments pertaining to the file. Parts of this record may be blank.
3	Data descriptor record	Describes the type of data contained in this file; information includes the processor, number of data bytes, file type, image ID, and base address. Parts of this record may be blank.
4	Data record	Contains the set of binary data dumped. This record provides values for all data dumped from the spacecraft.

Table 4-2. Dump Abstract Record Format

Field Width	Field Description
Up to 80 characters	Abstract—Intended to describe the functional purpose of the file and other information; blank when generated by the MOC.

Table 4-3. Dump Identification Record Format

Field Width	Field Description
Up to 15 characters	Mission name—LANDSAT7.

17 characters	Date created—The date the file was created, formatted as yyyy/ddd:hh:mm:ss.
3 characters decimal	Version number—Identifies the version of the file; zero-filled decimal format ("000" when generated by the MOC) or may be blank.
Up to 10 characters	Source—Identifies the originator (MOC) of the source file.
4 characters hexadecimal	File checksum—Zero-filled hexadecimal format ("0000" when generated by the MOC) or may be blank.

Table 4-4. Dump Data Descriptor Record Format

Field Width	Field Description
Up to 4 characters	Processor—Identifies which processor contains the data associated with this file. Possible values are SCP1 or SCP2 for standard controls processors 1 or 2, TDF for TDF dumps, USER for user dumps, or blank if unknown.
8 characters hexadecimal	Number of bytes—Indicates the total number of bytes in the file; represented in zero-filled hexadecimal.
4 characters	File type—Indicates the file is a dump image. This field should contain DUMP.
8 characters	Image identifier—Provides a mnemonic for the dump image. This field should contain one of the IDs defined by the image ID definition file received from the maintenance elements; or "TDF" for telemetry data formatter dumps, "USER" for user-defined dumps, or blank if unknown or mixed.
8 characters hexadecimal	Base address—Indicates the base address for the data contained in the file (i.e., an absolute address relative to the beginning of the memory area in the given SCP); represented in zero-filled hexadecimal. Maximum of "0002FFFE" for an SCP dump, "00000FFF" for a TDF dump, and "FFFFFFFF" for unknown or user-defined dumps.
6 characters	Memory type—Indicates the type of memory in which the data reside, either read-only memory (ROM), random access memory (RAM), electrically erasable programmable ROM (EEPROM), telemetry data formatter (TDF), user (USER), or blank if unknown.
4 characters hexadecimal	Data checksum in zero-filled hexadecimal format ("0000 when generated by the MOC).

Table 4-5. Dump Data Record Format

Field Width	Field Description
Twice the number of characters as the number of data bytes	Data bytes—Contains the data bytes that were dumped from spacecraft memory in binary representation.

Record 1 - Abstract record (ASCII, always blank):

<LF>

Record 2 - Identification record (ASCII):

LANDSAT7,1999/123:12:26:15,000,MOC,0000<LF>

Record 3 - Data descriptor record (ASCII):

SCP1,00000068,DUMP,ADFKDROM,00000584,ROM,0000<LF>

Record 4 - Data record (binary).

Figure 4-1. Example of a Dump Image File

4.2.2 Star Catalog Files

Updates to the star catalog are generated by the FDF as needed. The updates are transferred from the FDF to the MOC. The MOC transfers the updates generated by the FDF to the FSME.

The star catalog delivered to the FSME is in the same format as the star catalog file that the FDF sends to the MOC (i.e., the MOC does not modify the original FDF file before sending the star catalog to the FSME). The file format is described in the Landsat 7/FDF ICD (Reference 2). The MOC renames the FDF-generated star catalog file before transferring it to the FSME. The file-naming conventions for the star catalog files sent from the MOC to the FSME are provided in Appendix A.

4.2.3 Load Image File Checksum Error Notification

The MOC notifies the FSME by voice in the event the MOC detects a checksum error in a load image file (see Section 4.3.2).

4.3 FSME to MOC

4.3.1 Image ID Definition Files

The MOC receives an initial image ID definition file from the FSME before Landsat 7 launch. The FSME sends a new image definition to the MOC whenever the set of valid image IDs changes. The image ID definition file is an ASCII file that defines the allowable IDs for memory images that the MOC recognizes in dealing with the FSME. The file lists the names of all the partitions of memory images.

The file format for the image ID definition file is shown in Table 4-6. The image ID definition file consists of a header record followed by a variable number of data records. The header record contains any text comments about the file that the FSME chooses to include. The data records contain the image IDs (or memory partition names), the type of memory associated with the image ID, and the specified address locations for the image ID. Image ID records cannot overlap in address space. For a given memory address location, there can be no more than one image ID

Table 4-6. Image ID Definition File Format

Record Number	Record Type	Description
1	Header record	The first record in the file contains comments about the file. This record may be blank.
2	Data	The second record of the file starts a set of ASCII data records that define the set of image names, memory type associated with the image name, and valid addresses for each memory area partition.
3	Data	Same as record number 2 above.
N	Data	Same as record number 2 above.

that corresponds to that address location. Each record is terminated by a line feed character (ASCII 10). Fields within records are separated by ASCII commas. Fields are not blank filled to the maximum field width. Tables 4-7 and 4-8 describe the record header and data record field formats, respectively. The field widths shown in these tables do not include the terminating character. Figure 4-2 shows an example of the image ID definition file.

Table 4-7. Image ID Definition Header Record Format

Field Width	Field Description
Up to 40 characters	Comment—This record may be blank.

Table 4-8. Image ID Definition Data Record Format

Field Width	Field Description
8 characters	Image identifier—Provides the mnemonic name of the partition of memory.
Up to 6 characters	Memory type—Identifies the type of memory associated with the image ID (RAM, ROM, or EEPROM).
8 characters hexadecimal	Start address—Value for the partition of memory and represented in zero-filled hexadecimal. Range is “00000000” through “0002FFFF”.
8 characters hexadecimal	End address—Value for the partition of memory and represented in zero-filled hexadecimal. Range is “00000000” through “0002FFFF”.

```
LSC1997045IMAGE0.S02 image ID definition file<LF>
MMB00000, RAM, 00000000, 00001000<LF>
MME30000, ROM, 00001001, 00002000<LF>
RAM10000, EEPROM, 00004000, 00005000<LF>
ABCDEFGH, RAM, 00005001, 00006750<LF>
```

Figure 4-2. Example of an Image ID Definition File

The file-naming convention for the image ID definition file is provided in Appendix A.

4.3.2 Load Image Files

Load image files are sent to the MOC by the FSME as needed. A load image covers a partition or a contiguous subset of a partition associated with an image ID. Multiple load image files are required to load noncontiguous memory addresses or to load more than one partition.

The file format of the load image file is shown in Table 4-9. Load image files are ASCII files with ASCII header records. The load image file consists of three header records (abstract, identification, and data descriptor), one or more data records, and comment records. The abstract, identification, and the data descriptor header records contain information used for tracking purposes. The data records contain the actual data to be loaded to the spacecraft, and comment records are used for descriptive purposes.

Table 4-9. Load Image File Format

Record Number	Record Type	Description
1	Abstract record	Describes the functional purpose of the file and other information, such as the operational impact of loading this file. This record may be blank.
2	Identification record	Contains basic information that uniquely identifies this file; the information includes mission, creation date, version number, and source, separated by commas.
3	Data descriptor record	Describes the type of data contained in the file, including the processor, image ID, number of data bytes, and checksum of the file.
4	Data record	Starts a set of ASCII data records that provides values for all static data. In addition, data may be provided for all or part of the dynamic portion of the image.
*	Data record	Same as record number 4 above.
n	Data record	Same as record number 4 above

The ASCII header records are broken into fields. Each field, except for the last, is terminated by a comma; the last field in a record is terminated by a line-feed character (ASCII 10).

Each data record starts with the character X, indicating hexadecimal characters, followed by only the data bytes. Data bytes can have any number of blank characters in them for readability, and these blank characters are ignored. Data records are in ascending order by start location. The bytes in the data records are to be in the order loaded onboard the spacecraft. Each data record can contain up to 40 bytes of data. Each data record is terminated with an ASCII line-feed character (ASCII 10). A comment field may be attached to the end of a data record by

placing a semicolon in the record to start the comment. The semicolon, and anything after the semicolon and before the line feed, is ignored.

Comment records start with a semicolon with text following. They are terminated by a line-feed character (ASCII 10). Comment records can appear anywhere within the file.

Tables 4-10 through 4-12 describe the header record formats. Table 4-13 describes the data record format. Figure 4-3 shows an example of the load image file. The file-naming convention for the load image file is provided in Appendix A.

Table 4-10. Abstract Record Format

Field Width	Field Description
Up to 80 characters	Abstract—Specifies the functional purpose of the image. This record may be blank.

Table 4-11. Identification Record Format

Field Width	Field Description
Up to 15 characters	Mission name—LANDSAT7.
17 characters	Date created—Date the file was created, formatted as yyyy/ddd:hh:mm:ss.
3 characters decimal	Version number—Identifies the version of the file (in decimal and zero-filled).
Up to 10 characters	Source—Identifies the originator of the source file. Should be set to FSME.
4 characters hexadecimal	File checksum—Zero-filled hexadecimal format. Calculated by adding all ASCII bytes in the file from after the line feed (ASCII 10) that terminates the ID record. Line feeds and end-of-file markers are not included in the calculations.

Table 4-12. Data Descriptor Record Format

Field Width	Field Description
4 characters	Processor—Identifies which processor will be loaded with the data in the file (SCP1 or SCP2).
8 characters hexadecimal	Number of bytes—Indicates the number of actual data bytes to be uploaded into the SCP's memory; represented in zero-filled hexadecimal. Must be an even value.
4 characters	File type—Must be PART.
8 characters	Image ID—Must correspond to an image ID received in the image ID definition file.
8 characters	Base address—Indicates the base address for the data contained in the file (i.e., an absolute address relative to the beginning of the memory area in the given SCP); represented in zero-filled hexadecimal.
6 characters	Memory type—Indicates the type of memory in which the data reside, either ROM, RAM, or EEPROM.

4 characters hexadecimal	Data checksum in zero-filled hexadecimal format. Calculated by adding all data words in the file, ignoring all comments, line-feeds, and end-of-file markers.
--------------------------	---

Table 4-13. ASCII Data Record Format

Field Width	Field Description
Up to n characters hexadecimal	ASCII data bytes—Provides the data bytes to be loaded into spacecraft memory. Each data record starts with the character X. All data are in hexadecimal format. A semicolon indicates the beginning of a comment, and all data from the semicolon until the end of the line are ignored. No more than 40 bytes of data per record. Each data record must contain an even number of data bytes.

Record 1 - Abstract record (ASCII):

Memory Load to filter ACS<LF>

Record 2 - Identification record (ASCII):

LANDSAT7,1999/123:12:26:15,003,FSME,6CD4<LF>

Record 3 - Data descriptor record (ASCII):

SCP1,00000200,PART,RAM14500,00000000,EEPROM,12AB<LF>

Record 4 - Data record (ASCII):

```
; <LF>
X0002 1ED4 000259 00 180B C000 00B3 702A<LF>
; <LF>
X0003 ED34 00 164567 124F 3426 772F; A Comment<LF>
; <LF>
X0004 1ED4 002531 00 1808B234 C000 0001 000E 6A29<LF>
```

Figure 4-3. Example of a Load Image File

Section 5. Session Layer

5.1 General

The session layer provides system-dependent, process-to-process communications functions, which include

- Receipt and processing of incoming and outgoing logical link connect, disconnect, and abort requests
- Receipt and processing of incoming and outgoing data
- Detection of network disconnects and failure of the transport layer to deliver data in a timely manner

FTP is the Internet standard, high-level protocol for transferring files from one machine to another. The server side requires a client to supply a login ID and password before it honors file transfer requests. This layer complies with the FTP standard as specified in the Internet request for comment (RFC).

5.2 MOC to FSME Transmissions

The MOC uses FTP to transfer the dump image and star catalog files from the MOC to the FSME. The MOC employs the FTP PUT command to place the files in a designated directory on the FSME computer. If the files cannot be transferred, the FOT notifies the FSME personnel by telephone.

The backup means of file transfer will be 3.5-inch, double-sided, high-density disks.

5.3 FSME to MOC Transmissions

The FSME uses FTP to transfer the image ID definition and load image files from the FSME to the MOC. The FSME employs the FTP PUT command to place the files in a designated directory on the MOC computer. If the files cannot be transferred, FSME personnel notify the MOC FOT by telephone.

The backup means of file transfer will be 3.5-inch, double-sided, high-density disks.

Section 6. Transport Layer

The transport layer provides a system-independent, process-to-process communications service in association with the underlying services provided by the lower layers. The transport layer permits two systems to exchange data reliably and sequentially, regardless of their location within a network.

TCP is the standard transport level protocol that provides the reliable, full duplex, stream service on which many application protocols depend. TCP allows a process on one machine to send a stream of data to a process on another. Before transmitting data, participants must establish a connection. This layer complies with the TCP standard as specified in the Internet RFC.

Section 7. Network Layer

The network layer provides transparent data transfer between two transport layer entities. The network layer accepts packets from the transport layer at the source node and forwards them to the destination node.

IP is the Internet standard protocol that defines the Internet datagram as the unit of information passed across the Internet and provides the basis for the Internet connectionless, best-effort packet delivery service. This layer complies with the IP standard as specified in the Internet RFC.

Section 8. Data Link Layer

The data link layer creates the communications path between adjacent nodes and ensures the integrity of the data transferred between them. Functions covered by this layer include

- Establishing and terminating the link
- Detecting and responding to link transmission errors
- Synchronizing link data transmissions and reporting link status

The protocol governing this layer is the standard for the transmitting IP datagrams over Ethernet networks. The IP datagram is the basic unit of information passed across the Internet. It contains a source and destination address along with data. This layer complies with the IP datagram Internet standards as specified in the Internet RFCs.

Section 9. Physical Layer

The physical layer manages the physical transmission of data over a channel, which includes

- Monitoring change signals
- Handling hardware interrupts
- Informing the data link layer when transmission is complete

The MOC uses a standard 802.3 LAN connected to MODNET/NOLAN. Nascom provides connections between the MODNET/NOLAN and the FSME network for MOC and FSME electronic communications.

Appendix A. File-Naming Conventions

This appendix presents the file-naming convention for each product transferred between the MOC and the FSME:

File	From	To
Load image file	FSME	MOC
Dump image file	MOC	FSME
Image ID definition file	FSME	MOC
Star catalog file	MOC (through the FDF)	FSME
Supplementary star catalog file	MOC (through the FDF)	FSME

Load Image File

LSCyyydddnFSIMG.Svv

where yyyy = 4-digit year
 ddd = 3-digit day of year
 n = SCP designator (1 or 2)
 vv = version (within given day)

Dump Image File

LyyydddhmmssXD00000.Svv

where yyyy = 4-digit year
 ddd = 3-digit day of year
 hhhmmss = time in hours, minutes, and seconds
 X = an alphanumeric character that must be unique within a set of files having the same timestamp. Precise interpretation is TBR
 vv = version (within given day)

Image ID Definition File

LSCyyyddIMAGID.Svv

where yyyy = 4-digit year
 ddd = 3-digit day of year
 vv = version (within given day)

Star Catalog File

LSCyyydddnFSSTR.Svv

where yyyy = 4-digit year
 ddd = 3-digit day of year

n = SCP designator (1 or 2)
vv = version (within given day)

Supplementary Star Catalog File

LSCyyyydddSTRSUP.Svv

where yyyy = 4-digit year
 ddd = 3-digit day of year
 vv = version (within given day)

Abbreviations and Acronyms

ASCII	American Standard Code for Information Interchange
CCB	Configuration Control Board
COTS	commercial off-the-shelf
DCN	document change notice
EEPROM	electrically erasable programmable read-only memory (PROM)
ETM+	Enhanced Thematic Mapper Plus
FDF	Flight Dynamics Facility
FOT	Flight Operations Team
FSME	Flight Software Maintenance Element
FTP	File Transfer Protocol
GSFC	Goddard Space Flight Center
ICD	interface control document
ID	identifier
IP	Internet Protocol
ISO	International Standards Organization
LAN	local area network
LSIM	Landsat 7 simulator
MAC	media access connection
MO&DSD	Mission Operations and Data Systems Directorate
MOC	Mission Operations Center
MODNET	MO&DSD Operational/Development Network
MOSDD	Mission Operations and Systems Development Division
NASA	National Aeronautics and Space Administration
Nascom	NASA Communications
NOAA	National Oceanic and Atmospheric Administration
NOLAN	Nascom Operational Local Area Network
OSI	Open Systems Interconnection

PROM	programmable read-only memory
RAM	random access memory
RFC	request for comment
ROM	read-only memory
SCP	standards control processor
SDVF	Software Development and Validation Facility
SSR	solid state recorder
STE	special test equipment
TCP	Transmission Control Protocol
TDF	telemetry data formatter